CLAIMS

1.   A method of decoding a Turbo encoded code segment, comprising:

forming a plurality of hypotheses for the code segment, wherein each hypothesis corresponds to a particular set of one or more values for a set of one or more parameters used for decoding the code segment;

decoding the code segment in accordance with each of the hypotheses;

evaluating one or more performance metrics for a decoded result for each of the hypotheses;

determining a particular hypothesis having a best decoded result based on the one or more evaluated performance metrics; and

providing a sequence of decoded bits for the hypothesis with the best decoded result.

2.   The method of claim 1, wherein the decoding is performed for a particular number of iterations for each of the hypotheses, and wherein the one or more performance metrics are evaluated for each hypothesis after completing the particular number of iterations.

3.   The method of claim 2, wherein the particular number of iterations is less than that required to completely decode the code segment.

4.   The method of claim 1, wherein the decoding is performed based on a maximum *a posteriori* (MAP) decoding scheme.

5.   The method of claim 4, wherein the MAP decoding scheme utilizes a function for decoding the code segment, and wherein the set of one or more parameters includes at least one parameter for the function.

6.   The method of claim 5, wherein the function is a min* function.

7.   The method of claim 5, wherein the function is implemented in part with a set of lookup tables corresponding to different values for the function parameter.

8.   The method of claim 1, wherein the set of one or more parameters includes a parameter for a sequence of scaling factors used to scale bits received for the code segment to derive bits to be decoded.

9.   The method of claim 8, wherein the plurality of hypotheses correspond to a plurality of sequences of scaling factors.

10.   The method of claim 5, wherein the set of one or more parameters includes a parameter for a scale used to evaluate the function.

11.   The method of claim 10, wherein the plurality of hypotheses correspond to a plurality of scale values.

12.   The method of claim 8, wherein the set of one or more parameters further includes a parameter for a scale used to evaluate a function for decoding the code segment, and wherein the plurality of hypotheses correspond to a plurality of combinations of sequences of scaling factors and scale values.

13.   The method of claim 1, wherein the one or more performance metrics include a CRC check.

14.   The method of claim 1, wherein the one or more performance metrics include a correlation metric.

15.   The method of claim 1, wherein the one or more performance
metrics include a re-encoded correlation metric.

16.   The method of claim 1, wherein the one or more performance
metrics include a minimum or average log likelihood ratio (LLR) of the decoded bits.

17.   A method of decoding a Turbo encoded code segment,
comprising:

forming a plurality of hypotheses for the code segment,
wherein each hypothesis corresponds to a particular set of one or more values
for a set of one or more parameters used for decoding the code segment;

decoding the code segment based on a maximum *a posteriori*
(MAP) decoding scheme and in accordance with each of the hypotheses;

evaluating one or more performance metrics for a decoded
result for each of the hypotheses;

determining a particular hypothesis having a best decoded
result based on the one or more evaluated performance metrics; and

providing a sequence of decoded bits for the hypothesis with
the best decoded result, and

wherein the set of one or more parameters includes a parameter for a
scale for a min* function used to implement the MAP decoding scheme.

18.   A Turbo decoder comprising:

a constituent decoder configured to receive and decode bits for a code segment based on a particular decoding scheme and in accordance with a particular hypothesis to provide a decoded result for the hypothesis; and

a performance metric calculator configured to evaluate one or more performance metrics for the decoded result for the hypothesis, and

wherein a plurality of hypotheses are formed for the code segment, wherein each hypothesis corresponds to a particular set of one or more values for a set of one or more parameters used for decoding the code segment, and wherein the constituent decoder and performance metric calculator operate on the code segment for each of the hypotheses.

19.   The Turbo decoder of claim 18, wherein the constituent decoder is a maximum *a posteriori* (MAP) decoder.

20.   The Turbo decoder of claim 19, further comprising:

a plurality of lookup tables configured to implement a function for the MAP decoder, wherein each lookup table corresponds to a particular scale value for the function, and

wherein the plurality of hypotheses correspond to a plurality of scale values to be used for decoding the code segment.

21.   The Turbo decoder of claim 20, wherein the plurality of lookup tables are configured to implement a min* function for the MAP decoder.

22.   The Turbo decoder of claim 18, wherein the bits to be decoded are derived by scaling received bits with a sequence of scaling factors, and wherein the plurality of hypotheses correspond to a plurality of sequences of scaling factors.